

## **Pflichtenheft**

### **NetChat – Eine Multi-User Chat-Anwendung**

**Modul:** Patterns and Frameworks – Wintersemester 2025/26

**Team:** Waldemar Hermann, Joon-Hye Suk, Ines Güssow, Jasmin Gonser

**Betreuer:** Prof. Dr. sc. techn. Harald Loose

**Hochschule:** Technische Hochschule Brandenburg

## **Inhaltsverzeichnis**

1. Zielbestimmung	1
1.1 Grenzkriterien	1
1.2 Abgrenzungskriterien	2
2. Produkteinsatz	2
3. Produktfunktionen	2
4. Entwurfsmuster	3
5. Benutzeroberfläche	3
6. Datenverwaltung	3
7. Qualitätsziele	4
8. Entwicklungsumgebung	4
9. Projektplan	4
10. Fazit	5

## **1 Zielbestimmung**

Im Rahmen des Moduls *Patterns and Frameworks* soll eine Chat-Anwendung mit Mehrbenutzerfunktionalität entstehen.

Das Projekt trägt den Namen NetChat und soll zeigen, wie mit Hilfe von Softwarearchitektur, Entwurfsmustern und nebenläufiger Programmierung eine verteilte Anwendung umgesetzt werden kann.

Ziel ist es, eine stabile und benutzerfreundliche Kommunikationsplattform zu entwickeln, die gleichzeitig als Lehrbeispiel für gute Programmierarchitektur dient.

### **1.1 Grenzkriterien**

NetChat ist eine Echtzeit-Chat-Anwendung, die es mehreren Nutzern ermöglicht, über ein lokales Netzwerk oder das Internet miteinander zu kommunizieren.

Jede Benutzerin und jeder Benutzer kann sich über einen Client beim Server anmelden und Nachrichten an andere Teilnehmende senden.

Der Server verwaltet alle verbundenen Clients und verteilt die Nachrichten an die jeweiligen Empfänger.

#### Rollenmodell:

Benutzer (Client): Meldet sich mit einem Namen an, sendet und empfängt Nachrichten.

Serveradministrator: Startet und verwaltet den Server, überwacht Verbindungen und kann das System bei Bedarf beenden.

Kommunikation: Die Kommunikation erfolgt über TCP-Sockets. Jede Nachricht wird in einem definierten Format (z. B. JSON) übertragen.

#### Nebenläufigkeit:

Server: Thread pro Client-Verbindung

Client: Empfangsthread + JavaFX Application Thread

## **1.2 Abgrenzungskriterien**

- Keine Video- oder Sprachanrufe
- Kein Cloud- oder Online-Account-System
- Keine Werbung oder Monetarisierung

## **2 Produkteinsatz**

Anwendungsbereich:

Die Anwendung dient als Kommunikationsplattform für lokale Netzwerke (z. B. Campus, WG, Büro) und wird zu Demonstrationszwecken im Modul Patterns and Frameworks entwickelt.

Zielgruppen:

Studierende, Mitarbeitende oder Gruppen, die einfache textbasierte Kommunikation in Echtzeit wünschen.

Betriebsbedingungen:

- Java Runtime Environment (JDK 21 oder höher)
- Internet- oder LAN-Verbindung
- Lokaler Server auf PC oder Laptop

## **3 Produktfunktionen**

Benutzer können sich mit einem Namen anmelden, Nachrichten senden und empfangen, Chatverläufe einsehen und sich wieder abmelden.

Der Server verwaltet alle aktiven Clients und verteilt eingehende Nachrichten. Nachrichten werden lokal gespeichert (z. B. in SQLite oder JSON).

Hauptfunktionen:

- Verbindung herstellen
- Nachricht senden
- Nachricht empfangen
- Benutzerliste anzeigen
- Chatverlauf speichern

- Anwendung beenden

## 4 Entwurfsmuster

<b>Entwurfsmuster:</b>	<b>Verwendung:</b>
<b>MVC</b>	Strukturierung der Client-Architektur
<b>Observer</b>	UI aktualisiert sich automatisch bei neuen Nachrichten
<b>Singleton</b>	Zentrale Serverinstanz
<b>Factory</b>	Erzeugung von Nachrichtenobjekten oder Verbindungen
<b>Repository (optional)</b>	Verwaltung lokaler Chatverläufe

## 5 Benutzeroberfläche

Login-Fenster:

- Eingabe: Server-IP, Benutzername
- Button: „Verbinden“

Hauptfenster:

- Nachrichtenanzeige (Scrollbereich)
- Textfeld zur Eingabe
- „Senden“-Button
- Benutzerliste (rechts)
- Statusanzeige (Serververbindung, Benutzeranzahl)

## 6 Datenverwaltung

- Speicherung der Chatverläufe lokal in einer SQLite-Datenbank oder JSON-Datei
- Tabellen:
  - users (id, name)
  - messages (id, sender, text, timestamp)
- Server-Log für Systemmeldungen und Fehlermeldungen

## 7 Qualitätsziele

Ziel:	Beschreibung:
<b>Übersichtlichkeit</b>	Klare, einfache Benutzeroberfläche
<b>Stabilität</b>	Robuste Verbindung ohne Abstürze
<b>Performanz</b>	Nachrichtenzustellung < 1 Sekunde
<b>Fehlertoleranz</b>	Automatischer Reconnect bei Abbruch
<b>Plattformunabhängigkeit</b>	Lauffähig auf Windows, Linux und macOS

## 8 Entwicklungsumgebung

- **Programmiersprache:** Java (JDK 21)
- **Frameworks:** JavaFX, SQLite
- **IDE:** IntelliJ IDEA oder Eclipse
- **Build-System:** Maven oder Gradle
- **Versionsverwaltung:** GitHub
- **Betriebssystem:** Windows, macOS oder Linux

## 9 Projektplan

### Arbeitspakete

ID	Inhalt	Zeitraum	Abhängigkeit
1	Projektstart & Einführung (Sprechstunde, Gruppenbildung)	30.09.–19.10.2025	–
2	Pflichtenheft erstellen & abstimmen	20.–26.10.2025	1
3	Pflichtenheft-Verteidigung (Online)	28.10.2025	2
4	Entwurf der Anwendung (GUI, Datenmodell, Architektur)	29.10.–01.12.2025	3
5	Zwischenpräsentation (Pflichtpräsenz)	02.12.2025 oder 06.01.2026	4
6	Implementierung – Phase 1 (Server, Netzwerk, Threads)	02.–20.12.2025	4
7	Implementierung – Phase 2 (Client, GUI, MVC-Struktur)	02.–22.12.2025	4
8	Implementierung – Phase 3 (Datenbank, Speicherung)	05.–30.12.2025	6, 7
9	Integration & Tests	03.–20.01.2026	8
10	Dokumentation & Präsentation	21.–25.01.2026	9
11	Endabgabe (App + Dokumentation)	25.01.2026	10
12	Abschlusspräsentation / Prüfung	<b>31.01.2026</b>	11

## Zeitplan (Übersicht)

<b>Phase</b>	<b>Meilenstein</b>	<b>Termin</b>	<b>Art</b>
<b>1</b>	Einführungssprechstunde	30.09.2025, 20 Uhr	online
<b>2</b>	Gruppenbildung & Themenwahl	bis 19.10.2025	per Mail
<b>3</b>	Pflichtenheft – Abgabe	26.10.2025	online
<b>4</b>	Pflichtenheft – Verteidigung	28.10.2025	online
<b>5</b>	Zwischenpräsentation	02.12.2025 oder 06.01.2026	online
<b>6</b>	Abgabe Anwendung & Doku	25.01.2026	Abgabe
<b>7</b>	Präsentation / Prüfung	<b>31.01.2026</b>	Präsenz / online

## 10 Fazit

NetChat ist eine verteilte Chat-Anwendung, die alle zentralen Themen des Moduls Patterns and Frameworks praxisnah abbildet.

Sie demonstriert Entwurfsmuster (MVC, Observer, Singleton), Softwarearchitektur, Nebenläufigkeit, UI-Design und Datenhaltung.

Das Projekt ist realistisch umsetzbar, im Team gut aufteilbar und eignet sich hervorragend für die Prüfung und Präsentation.